

**REMARKS**

This application has been carefully considered in connection with the Examiner's Final Office Action dated September 11, 2008. Reconsideration and allowance are respectfully requested in view of the following.

**Summary of Rejections**

Claims 1-23 were pending at the time of the Final Office Action.

Claims 1-23 were rejected under 35 USC § 102.

**Summary of Response**

Claims 1-8, 12, 13, 16-20, and 23 are currently amended herein.

Claim 15 is currently cancelled herein.

Claims 10, 11, and 14 remain as previously presented.

Claims 9, 21, and 22 remain as originally submitted.

Remarks and Arguments are provided below.

**Summary of Claims Pending**

Claims 1-14 and 16-23 are currently pending following this response.

**Response to Rejections**

Mullins does not anticipate a listener that simulates a driver of a data store to which a commercial-off-the-shelf software application is certified to communicate data requests. Additionally, Mullins does not anticipate that the translator translates data requests received by the listener into a generic format or a data access layer that determines where to direct the data requests and further translates the data request from the generic format into the format of the appropriate data store for fulfilling the data requests. Simulating the driver by the listener decouples the commercial-off-the-shelf application from the data stores that fulfill its data requests by enabling the commercial-off-the-shelf application to submit data requests in its normal manner regardless of the characteristics of the data stores that fulfill its data requests. Translating the data requests into the format of the appropriate data store for fulfilling the data requests enables data stores different from the data stores for which the applications were originally certified to fulfill the data requests. Therefore, the data stores that fulfill the data requests may be changed or upgraded as needed without the need for any changes in the commercial-off-the-shelf application.

Mullins teaches a translator that translates between a wide variety of different data formats and a database driver that accesses a database directly using the data format translated by the translator. Although Mullins discloses a driver, the driver of Mullins does not anticipate a listener that simulates a driver of a data store to which a commercial-off-the-shelf software application is certified to communicate data requests. Furthermore, while Mullins may disclose a translator, the translator of Mullins does not anticipate a translator that translates data requests received by the listener into a

generic format or a data access layer that determines where to direct the data requests and further translates the data requests from the generic format into the format of the appropriate data store for fulfilling the requests.

These and other distinctions between the pending specification and the applied art will be discussed in greater detail in the analysis of the pending claims that follows.

### **Response to Rejections under Section 102**

#### **Claim 1:**

Claim 1 was rejected under 35 USC § 102(e) as being anticipated by Mullins, U.S. Patent No. 7,149,730 (hereinafter "Mullins").

I. Mullins does not anticipate a listener that simulates a driver of a database for receiving a data request from a commercial-off-the-shelf application intended for the database.

Amended Claim 1 recites:

a plurality of drivers, wherein each of the plurality of the first data stores and the plurality of second data stores has a corresponding one of the plurality of drivers configured to receive the data request and pass the data request to the corresponding data store . . . a listener . . . [that] simulates one of the plurality of drivers corresponding with one of the plurality of first data stores and receives the data request from one of the plurality of commercial-off-the-shelf software applications that is compatible with the one of the plurality of first data stores simulated by the listener

Applicants respectfully submit that no new matter has been introduced by amendments to Claim 1. Support may be found throughout the specification as originally filed, including paragraphs 19-22, 24, 27, 28, and 30.

In reference to a rejection of independent Claim 1, the Final Office Action alleges on pages 2 and 3 that:

Mullins discloses . . . a listener . . . to receive the output from the commercial-off-the shelf software application (col. 4:40-50 ". . . access the database directly through a database driver . . ." and col. 12:45-50 ". . . using a JDBC driver to which is passed data store access code 205 of Figure 5" and col. 27:15-30 and e.g. Figure 7, Abstraction Layer 1 and related text)

Based on the disclosure cited above, Mullins teaches that the JDBC driver, which the Final Office Action equates to the listener, directly accesses a data store. In contrast, the pending disclosure teaches a listener that receives the data request from the commercial-off-the-shelf software application, but that does not communicate directly with the data store. In contrast to the Final Office Action's equation of the pending disclosure's listener to Mullins' data store driver, paragraphs 19, 20, and 21 of the pending disclosure clearly differentiate between the listener of the pending disclosure and the data store driver of Mullins. Paragraphs 19, 20, and 21 of the pending disclosure are reproduced below:

The listeners 32, 34, and 36 simulate the functions of the listeners that are typically built in to commercially available data stores. More specifically, each listener 32, 34, or 36 simulates the listener of the data store for which its corresponding COTS application 12, 14, or 16 was certified. Thus, a COTS application 12, 14, or 16 can submit a data request in its normal manner and, from its perspective, the data request is received by the same listener that normally receives its data requests.

Each listener 32, 34, or 36 communicates with a corresponding translator 42, 44, or 46. The translators 42, 44, and 46 translate data requests between the formats of the listeners 32, 34, and 36 and a single or generic format . . .

the Oracle 9 database 92 has an Oracle 9 driver 82 and an Oracle 9 wrapper 72, the DB2 database 94 has a DB2 driver 84 and a DB2 wrapper 74, and the Teradata database 96 has a Teradata driver 86 and a Teradata wrapper 76.

In contrast to Mullins' driver, which receives the translated data request and communicates directly with the data store, the listener taught by the pending disclosure simulates a driver of a database for receiving a data request from a commercial-off-the-shelf application intended for the database and conveys the received data request to a translator and is only conveyed to the data store. Moreover, Applicants note that a text search of the disclosure of Mullins for the term "simulate" only produced one result in the table shown in column 42 of Mullins. Applicants respectfully submit that Mullins disclosure in column 42 of simulating a primary key of a database is not explicit or inherent disclosure of a listener that simulates a driver of a database for receiving a data request from a commercial-off-the-shelf application intended for the database.

Accordingly, Mullins does not anticipate a listener that simulates a driver of a database for receiving a data request from a commercial-off-the-shelf application intended for the database.

II. Mullins does not anticipate a translator that receives a data request from a listener, translates the data request, and submits the translated data request to a driver for storage by a data store.

Amended Claim 1 recites:

a plurality of drivers, wherein each of the plurality of the first data stores and the plurality of second data stores has a corresponding one of the plurality of drivers configured to receive a data request and pass the data request to the corresponding data store . . . a translator . . . [that] receives the data request from the listener, translates the data request, and submits the translated data request for one of the plurality of drivers corresponding with one of the plurality of second data stores for storage by the one of the plurality of second data stores.

Applicants respectfully submit that no new matter has been introduced by amendments to Claim 1. Support may be found throughout the specification as originally filed, including paragraphs 19-22, 24, 27, 28, and 30.

In reference to a rejection of independent Claim 1, the Final Office Action alleges on pages 2 and 3 that:

Mullins discloses . . . a translator . . . in communication with the listener . . . to receive the output and . . . translate the output . . . for storage by one of the plurality of second data stores (col. 8: 1-15 ". . . translate data between each of an object database, a relational database . . .")

Based on the disclosure cited above, Mullins teaches a translator that translates between a wide variety of different data formats and a database driver that accesses the database directly using the data format translated by the translator. In contrast, the pending disclosure teaches a translator that receives the data request from the commercial-off-the-shelf software application after the listener receives the data request from the commercial-off-the-shelf software application. Claim 2, Claim 13, and Claim 17 limitations clarify that after the translator translates the data request into a generic format, a data access layer determines where to direct the translated data request, and translates the data request from the generic format into a storage format based on this determination. Claims 2, 13, and 17 further clarify that a wrapper receives and wraps the translated data request, a different driver receives the wrapped and translated data request, and only then is the data store accessed to store the wrapped and translated data request. As discussed above in section I, the listener from which the translator receives the data request differs from the database driver which subsequently receives the translated data request.

Therefore, the translator taught by the pending disclosure receives an un-translated data request from the listener that simulates a database driver, translates the data request, and submits the translated data request to a different database driver. In contrast, Mullins teaches a translator in communication with a database driver that the Final Office Action equates to the pending disclosure's listener. Mullins does not anticipate that a database driver provides the translator with an un-translated data request that is to be translated and submitted to the same or different database driver. Therefore, Mullins does not anticipate a translator that receives a data request from a listener, translates the data request, and submits the translated data request to a driver for storage by a data store.

For at least the reasons established above in sections I and II, Applicants respectfully submit that independent Claim 1 is not anticipated by Mullins and respectfully requests allowance of this claim.

**Claims depending from Claim 1:**

Claims 2-12 were rejected under 35 USC § 102(e) as being anticipated by Mullins.

Dependent Claims 2-12 depend directly or indirectly from independent Claim 1 and incorporate all of the limitations thereof. Accordingly, for at least the reasons established in sections I and II above, Applicants respectfully submit that Claims 2-12 are not anticipated by Mullins and respectfully requests allowance of these claims.

III. Mullins does not anticipate a translator that translates a data request into a generic format and a data access layer that determines to direct the data request from a commercial-off-the-shelf software application to a second data store, and to translate the translated data request from the generic format into a storage format of the second data store.

Amended Claim 2 recites

wherein the translator translates the data request into a generic format, and further comprising a data access layer . . . [that] determines to direct the data request from one of the commercial-off-the-shelf software applications to the one of the plurality of second data stores, and translates the translated data request from the generic format into a storage format of the one of the plurality of second data stores.

Applicants respectfully submit that no new matter has been introduced by amendments to Claim 2. Support may be found throughout the specification as originally filed, including paragraph 20, 22, 25, and 28.

In reference to the rejection of independent Claim 1, upon which dependent Claim 2 directly depends, page 3 of the Final Office Action cites column 29, lines 20-30, and column 8, lines 1-15, to allege that Mullins anticipates a translator to translate the output. Although these cited portions of Mullins mention translating, these portions do not identify any specific element of Mullins as actually translating data, nor do these portions mention how translating is performed. However, column 8, lines 66-67, of Mullins identifies the component that performs translating work, "The adapter abstraction layer 600 of the '197 patent performs any translation work necessary." Furthermore, in reference to the rejection of dependent Claim 2, page 4 of the Final



Office Action also equates the data access layer of the pending disclosure to the same adapter abstraction layer:

Mullins discloses . . . a data access layer . . . to determine where to direct the output from one of the commercial-off-the-shelf software application to one of the plurality of second data stores (e.g. Figure 1, 600 and col. 8:50-60 " . . . an abstract layer . . . data stores . . . " and col. 9:1-10 " . . . adapter abstraction layer 600 of Figure 1 . . . " and col. 13:30-40 " . . . specific adapter and data object . . . specific data stores supporting appropriate reference handling . . .").

Mullins offers an example of how the adapter abstraction layer actually performs:

the request 100 . . . was passed from the object application 101 to the adapter abstraction layer 600 comprising the first adapter 400. The first adapter 400 then extracted the object attributes 103 and the object name 104 from the object 102, and packed the object attributes 103 and the object name 104 as data 105 to be used in communication and transport layers. The first adapter 400 then communicated the data 105 and the request 100 to the second adapter 500 . . .

The second adapter 500 searched a meta data map 206 comprising at least one object name 114 using the object name 104 to determine whether the object name 104 existed in the meta data map 206. If the meta data map 206 contained the object name 104, then the second adapter 500 used the object attributes 103 and the found meta data map 201 to generate at least one command 303 for accessing the data store 302 according to the request 100 . . .

The second adapter 500 then executed at least one such command 303 as a data store access client server process 700 in communication with the data store 302 using a JDBC driver to which is passed data store access code 205 and obtains the data store content 301 and an execution status 306 based on executing at least one such command 303. (Column 12, lines 17-51)

The above portion of Mullins is partially cited in the Final Office Action in the rejection of each independent claim of the pending disclosure. Mullins teaches that a first adapter in the adapter abstraction layer extracts an object name and attributes from a request and communicates the extracted data to a second adapter, which uses the object name and the object attributes to generate and execute a command for accessing a data store via a driver. In contrast, the pending disclosure teaches a

translator that translates a data request into a generic format, and a data access layer that translates the data request from the generic format into the storage format used for the data store to which the data access layer is directing the translated data request. Extracting an object name and object attributes does not anticipate translating data request into a generic format. Using an extracted object name and object attributes to generate and execute a command to access a data store does not anticipate translating a data request from a generic format into the storage format specified for the data store.

In addition to the reasons established in sections I and II above, for at least the reasons established in section III above, Applicants respectfully submit that all of the limitations of dependent claim 2 are not anticipated by Mullins and respectfully request allowance of this claim.

**Claim 13:**

Claim 13 includes limitations substantially similar to the limitations discussed in sections I, II, and III above. For example, as amended herein, Claim 13 recites:

a first driver configured to receive the data request and pass the data request to the first data store. . . a listener . . . [that] simulates the first driver and receives the data request from the commercial-off-the-shelf software application submitted for the first driver; a translator . . . [that] receives the data request from the listener and translates the data request into a generic format to produce a first translated data request; a data access layer . . . [that] determines, based on an enterprise data model, to direct the data request of the commercial-off-the-shelf software applications to a second data store and translates the first translated data request from the generic format into a storage format of the second data store to produce a second translated data request . . . a second driver configured to receive the wrapped second translated data request and pass the wrapped second translated data request to the second data store

Applicants respectfully submit that no new matter has been introduced by this amendment. Support may be found throughout the specification as originally filed, including paragraphs 19-22, 24, 25, 27, 28, and 30. Accordingly, the arguments of sections I, II, and III are hereby repeated for Claim 13.

For at least the reasons established above in sections I, II, and III, Applicants respectfully submit that independent Claim 13 is not anticipated by Mullins and respectfully request allowance of this claim.

**Claims Depending from Claim 13:**

Claim 14 was rejected under 35 USC § 102(e) as being anticipated by Mullins.

Dependent Claim 14 depends directly from independent Claim 13 and incorporates all of the limitations thereof. Accordingly, for at least the reasons established in sections I, II, and III above, Applicants respectfully submit that Claim 14 is not anticipated by Mullins and respectfully request allowance of these claims.

**Claim 16:**

Claim 16 includes limitations substantially similar to the limitations discussed in sections I and II above. For example, as amended herein, Claim 16 recites:

a first driver configured to receive the data request and pass the data request to the first data store . . . a listener . . . [that] simulates the first driver and receives the data request from the commercial-off-the-shelf software application submitted for the first driver; a translator . . . [that] receives the data request from the listener and translates the data request; a second driver configured to receive the translated data request and pass the translated data request to the second data store

Applicants respectfully submit that no new matter has been introduced by this amendment. Support may be found throughout the specification as originally filed,

including paragraphs 19-22, 24, 27, 28, and 30. Accordingly, the arguments of sections I and II are hereby repeated for Claim 16.

For at least the reasons established above in sections I and II, Applicants respectfully submit that independent Claim 16 is not anticipated by Mullins and respectfully request allowance of this claim.

**Claims Depending from Claim 16:**

Claims 17-23 were rejected under 35 USC § 102(e) as being anticipated by Mullins.

Dependent Claims 17-23 depend directly or indirectly from independent Claim 16 and incorporate all of the limitations thereof. Accordingly, for at least the reasons established in sections I and II above, Applicants respectfully submit that Claims 17-23 are not anticipated by Mullins and respectfully request allowance of these claims.

Dependent Claim 17 includes limitations substantially similar to the limitations discussed in section III above. For example, as amended herein, Claim 17 recites:

wherein the translator translates the data request into a generic format, and further comprising a data access layer . . . [that] determines, based on an enterprise data model, to direct the data request from one of the commercial-off-the-shelf software applications to the second data store, and translates the translated data request from the generic format into a storage format of the second data store.

Accordingly, the arguments of section III are hereby repeated for Dependent Claim 17.

In addition to the reasons established in sections I and II above, for at least the reasons established in section III above, Applicants respectfully submit that all of the limitations of dependent Claim 17 are not anticipated by Mullins and respectfully request allowance of this claim.

**Conclusion**

Applicants respectfully submit that the present application is in condition for allowance for the reasons stated above. If the Examiner has any questions or comments or otherwise feels it would be helpful in expediting the application, he is encouraged to telephone the undersigned at (972) 731-2288.

The Commissioner is hereby authorized to charge payment of any further fees associated with any of the foregoing papers submitted herewith, or to credit any overpayment thereof, to Deposit Account No. 21-0765, Sprint.

Respectfully submitted,

Date: November 11, 2008

/Michael W. Piper/

Michael W. Piper

Reg. No. 39,800

CONLEY ROSE, P.C.  
5601 Granite Parkway, Suite 750  
Plano, Texas 75024  
(972) 731-2288  
(972) 731-2289 (facsimile)

ATTORNEY FOR APPLICANTS